Sebastian Neumaier and Axel Polleres

Geo-Semantic Labelling of Open Data. SEMANTiCS 2018-14th International Conference on Semantic Systems

Article (Published)
(Refereed)

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2018) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

SEMANTiCS 2018 – 14th International Conference on Semantic Systems

Geo-Semantic Labelling of Open Data

Sebastian Neumaier[a,1], Axel Polleres[a,b,c,2]

[a] *Vienna University of Economics and Business, Vienna, Austria*
[b] *Complexity Science Hub Vienna, Austria*
[c] *Stanford University, CA, USA*

Abstract

In the past years Open Data has become a trend among governments to increase transparency and public engagement by opening up national, regional, and local datasets. However, while many of these datasets come in semi-structured file formats, they use different schemata and lack geo-references or semantically meaningful links and descriptions of the corresponding geo-entities. We aim to address this by detecting and establishing links to geo-entities in the datasets found in Open Data catalogs and their respective metadata descriptions and link them to a knowledge graph of geo-entities. This knowledge graph does not yet readily exist, though, or at least, not a single one: so, we integrate and interlink several datasets to construct our (extensible) base geo-entities knowledge graph: (i) the openly available geospatial data repository GeoNames, (ii) the map service OpenStreetMap, (iii) country-specific sets of postal codes, and (iv) the European Union's classification system NUTS. As a second step, this base knowledge graph is used to add semantic labels to the open datasets, i.e., we heuristically disambiguate the geo-entities in CSV columns using the context of the labels and the hierarchical graph structure of our base knowledge graph. Finally, in order to interact with and retrieve the content, we index the datasets and provide a demo user interface. Currently we indexed resources from four Open Data portals, and allow search queries for geo-entities as well as full-text matches at `http://data.wu.ac.at/odgraph/`.

Keywords: open data; spatio-temporal labelling; spatio-temporal knowledge graph

1. Introduction

Open Data as a trend among governments is a relatively new but potentially powerful movement. The publishing agencies and organizations release local, regional and national data to a variety of users (citizens, businesses,

E-mail addresses: sebastian.neumaier@wu.ac.at (Sebastian Neumaier)., axel.polleres@wu.ac.at (Axel Polleres).

[1] Sebastian Neumaier's work was funded by the Austrian Federal Ministry of Transport, Innovation and Technology (BMVIT) through the program "ICT of the Future" by the Austrian Research Promotion Agency (FFG) under the project CommuniData.
[2] Axel Polleres' work was supported under the Distinguished Visiting Austrian Chair Professors program hosted by The Europe Center of Stanford University.

academics, civil servants, etc.). The data is collected as part of census collections, infrastructure assessments or any other, secondary output data; typical datasets, for instance, include public transport data of cities, results of past elections (split by subdivisions of a region/country), demographic indicators, etc. The idea and intention behind the open datasets is to increase public transparency [2] and to allow enhanced data-enriched citizen engagement in policy and other analysis [5, 10].

As most of the data is regional/national census-based, a large proportion of the resources available on governmental Open Data portals contain or refer to some kind of geographic information. There are several widely-used standards and structured file formats with agreed schemata to provide such geographic information in defined ways using points, lines, and polygons; for instance, the proprietary Shapefile format[4]: a binary geospatial vector data format, decidedly for GIS (geographic information system) software. Alternatively, the open standard GeoJSON[3] which is an extension of the JSON format.

However, geospatial information in Open Data, as it is published by governments and organizations, currently mainly still comes in semi-structured – however, *human-readable* – representations. Our 2016 quality report on 260 data portals [13] shows that no GIS file format is among the top ten most used formats across the portals and that semi-structured file formats such as tabular data in CSV or even in proprietary formats like XLS, and without predefined schema lead the list. Geo-references in these tabular sources are also not encoded structuredly or homogeneously, but using mixes of region names, country codes, or other implicit references. There are several possible reasons for this: (i) The target users of these datasets are non-experts (i.e., not familiar with GIS software/formats) and prefer human-readable labels for regions and places. (ii) The geographic information is missing for the data, i.e., the publishers do not have coordinates or location information, or (iii) the data provider lack the technical know-how to publish this geographic information.

From a Semantic Web viewpoint neither of the two approaches alone is fully satisfactory in order to integrate the datasets in the LOD cloud, where we want both, the structured unambiguous information and human readable labels linked alongside: Having human-readable labels for the locations in a dataset (e.g., regions, streets, or places) requires instance matching – using the appropriate ontology – in order to establish the desired links. Also, additional context information might be necessary in order to determine intended geo-entities; labels might be ambiguous, language-specific, or encoded using country-specific codes. On the other hand, having the rich geographic description is for sure the most comprehensive way of publishing data, however, we still miss the links to the corresponding geo-entities which provide additional description and information.

To close this gap between the currently published Open Data – where we find semi-structured formats, such as CSV, and human-readable descriptions (i.e., region names, places and streets) for geo-locations – and existing geo-data repositories and datasets, we propose Linked Data and Knowledge Graphs to the rescue and demonstrate that these ingredients contribute as follows:

1. We use the existing Linked Data repositories GeoNames, the map service OpenStreetMap, the European Union's classification scheme NUTS and country-specific postal codes datasets, and integrate these datasets in a hierarchical base geo-entities knowledge graph that is already linked to LOD.
2. Using this base knowledge graph we label metadata and data, i.e., meta-information of datasets available at Open Data portals and columns of CSVs.
3. In case of multiple labelling candidates (e.g., same street names in different cities/countries) we disambiguate the values based on the context, i.e. the surrounding values, by exploiting the graph structure of the base knowledge graph.
4. We index all the labelled datasets in a search engine to enable search functionality over geo-entities, but also to enable full-text search over the datasets. The search interface is available online at `http://data.wu.ac.at/odgraph/`

The remainder of this paper is structured as follows: In the following Section 2 we introduce (linked) datasets, repositories and endpoints to retrieve relevant spatial information. Section 3 describes the construction and integration of spatial-data repositories and endpoints into our base knowledge graph, and the algorithms to add geo-annotations to datasets from Open Data portals. In Section 4 we present the back-end, the user interface, and introduce the indexed data portals; we evaluate and discuss the performance (in terms of precision and recall based on a manually

generated sample) and limitations of our approach. We provide related and complementary approaches in Section 5, and eventually conclude in Section 6.

2. Background

The following section gives an overview of the GeoNames repository, OpenStreetMap, and available NUTS and postal codes datasets that we use and access in this work.

GeoNames. The GeoNames database[3] contains over 10 million geographical names of entities such as countries, cities, regions and villages, toghether with their alternative names in various languages. It assigns unique identifiers to geo-entities and provides a detailed hierarchical description including countries, federal states, regions, cities, etc.. For instance, the GeoNames-entity for the city of Munich[4] has the parent relationship "Munich, Urban District", which is located in the region "Upper Bavaria" of the federal state "Bavaria" in the country "Germany", i.e. the GeoNames database allows us to extract the following hierarchical relation for the city of Munich: *Germany* > *Bavaria* > *Upper Bavaria* > *Munich*, *Urban District* > *Munich*

The relations are based on the GeoNames ontology[5] which defines first-order administrative division (gn:A), second-order (gn:A.ADM2) , ... (until gn:A.ADM5)[6] for countries, states, cities, and city districts/sub-regions. In this work we make use of an RDF dump of the GeoNames database.[7]

OpenStreetMap (OSM). OSM[8] was funded in 2004 as a collaborative project to create free editable geospatial data. The map data is mainly produced by volunteers using GPS devices (on foot, bicycle, car, ..) and later by importing commercial and government sources, e.g., aerial photographies. Initially, the project focused on mapping the United Kingdom but soon was extended to a worldwide effort. OSM uses four basic "elements" to describe geo-information:[9]

- *Nodes* in OSM are specific points defined by a latitude and longitude.
- *Ways* are ordered lists of *nodes* that define a line. OSM ways can also define polygons, i.e. a closed list of nodes.
- *Relations* define relationships between different OSM elements, e.g., a *route* is defined as a relation of multiple ways (such as highway, cycle route, bus route) along the same route.
- *Tags* are used to describe the meaning of any elements, e.g., there could be a tag `highway=residential` (tags are represented as key-value pairs) which could be used on a *way* element to indicate a road within settlement.

There are already existing works which exploit the potential of OSM to enrich and link other sources. For instance, in [15] we have extracted indicators, such as the number of hotels or libraries in a city, from OSM to collect statistical information about cities.

Likewise, the software library *Libpostal*[10] uses OSM addresses and places: it provides street address parsing and normalization by using machine learning algorithms on top of the OSM data. The library converts free-form addresses into clean normalized forms and can therefore be used as a pre-processing step to geo-tagging of streets and addresses. We integrate Libpostal in our framework in order to detect and filter streets and city names in text and address lines.

Postal codes. These regional codes consist of a series of letters (not necessarily digits) with the purpose of sorting mail. Since postal codes are country-specific identifiers, and its granularity and availability strongly varies for different countries, there is no single, complete, data source to retrieve these codes. The most reliable way to get the complete

[3] http://www.geonames.org/
[4] http://www.geonames.org/6559171/
[5] http://www.geonames.org/ontology/ontology_v3.1.rdf
[6] Here, gn: corresponds to the namespace URL http://www.geonames.org/ontology#
[7] http://www.geonames.org/ontology/documentation.html, last accessed 2018-04-30
[8] https://www.openstreetmap.org/
[9] A detailed description can be found at the OSM documentation pages: http://wiki.openstreetmap.org/wiki/Main_Page
[10] https://medium.com/@albarrentine/statistical-nlp-on-openstreetmap-b9d573e6cc86, last accessed 2017-09-12

dataset is typically via governmental agencies (made easy, in case they publish the codes as open data).[11] Another source worth mentioning for matching postal codes is GeoNames: it provides a quite comprehensive collection of postal codes for certain countries and the respective name of the places/districts.[12]

Postal codes as Linked Data. Unfortunately, the above mentioned GeoNames postal codes dataset is neither fully included in the existing RDF version of the complete GeoNames dataset, nor in the available GeoNames-APIs for retrieving information about the geo-entities. Moreover, partially, postal codes for certain countries are available in the knowledge bases of Wikidata and DBpedia (see below) for the respective entries of the geo-entities (using "postal code" properties). However, we stress that these entries are not complete, i.e., not all postal codes are available in the knowledge bases as not all respective geo-entities are present, and also, the codes' representation is not standardized.

NUTS. NUTS (French: nomenclature des unitès territoriales statistiques) is a geocode standard developed and regulated by the European Union (EU). It references the subdivisions of all EU member states in three hierarchical levels, NUTS 1, 2, and 3. All codes start with the two-letter ISO 3166-1 [1] country code and each level adds an additional number to the code, cf. the NUTS hierarchy for Austria in Figure 1. The current NUTS classification lists 98 regions at NUTS 1, 276 regions at NUTS 2 and 1342 regions at NUTS 3 level.[13]

3. Approach

In the following we use the introduced geo-data repositories and dataset – some already available as Linked Data – to build up a base knowledge graph and to label Open Data resources: Section 3.1 discusses the integration of the datasets (using GeoNames as the basis), 3.2 displays the available cues for geo-information in data portals and 3.3 details the labelling algorithms for the datasets.

3.1. Building up a Base Knowledge Graph of Geo-Entities

Mapping of postal codes to GeoNames. The dataset of postal codes by GeoNames (cf. Section 2) consists of 84 countries and a total of 1.1M entries. For each code it provides a *place name*, and (depending on the country) several parent region/subdivion names. Based on these names we used a simple heuristic to map the postal codes to GeoNames entities: We preprocess the place name of the postal code by splitting on separators (white spaces, "-", "/")[14] and try to find GeoNames entries, in the respective country, with matching names. Then we use the code's parent regions to select the GeoNames entry (in case there are several candidates).

NUTS to GeoNames. Wikidata already includes several links to the GeoNames repository as well as a property for the NUTS classifications of regions, so we can use the Wikidata SPARQL endpoint[15] which provides mappings for 1316 (out of 1716) NUTS codes. The missing 400 codes are typically NUTS regions where no Wikidata and/or GeoNames entry exists because, strictly speaking, there is no such corresponding administrative region. For instance, the Austrian NUTS regions AT126 and AT127 are called "Wiener Umland/Nordteil" and "Wiener Umland/Südteil", however, these are no political districts, but they group a *set* of districts (lying north and south of the city of Vienna) and therefore there is no separate Wikidata/GeoNames entity to map.

OSM integration. To cover a more detailed and larger set of labels as it is available in GeoNames, e.g., the set of all street names and local places/POIs of a city, we extract OSM ways and nodes and map these to the GeoNames hierarchy. To do so we perform the following steps on a local extract of OSM:

[11] For instance, the complete list of Austrian postal codes is available as CSV via the Austrian Open Data portal: https://www.data.gv.at/katalog/dataset/f76ed887-00d6-450f-a158-9f8b1cbbeebf, last accessed 2018-01-05

[12] http://download.geonames.org/export/zip/, last accessed 2018-01-05

[13] http://ec.europa.eu/eurostat/web/nuts/overview, last accessed 2018-01-05

[14] We add this preprocessing step because there are many translated place names separated by slash or comma.

[15] https://query.wikidata.org/ with the following query to get these NUTS-to-GeoNames mappings: `SELECT ?s ?nuts ?geonames WHERE {?s wdt:P605 ?nuts.?s wdt:P1566 ?geonames}`

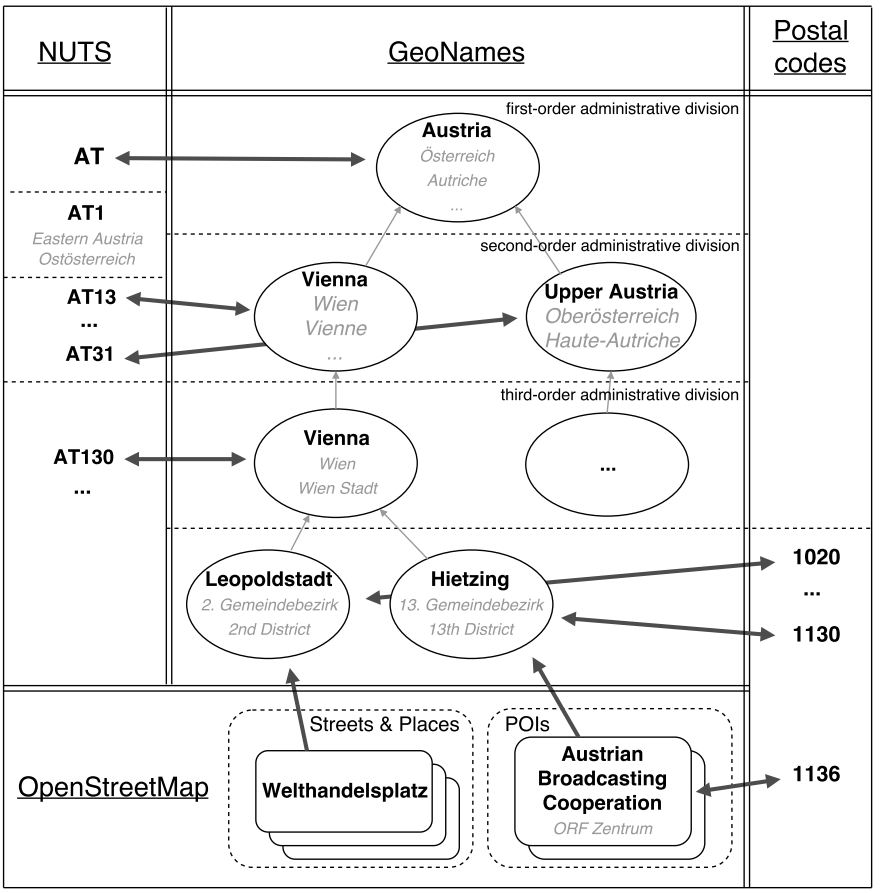

Fig. 1. Data integration: Bold arrows indicate established links and gray arrows the hierarchical structure of the geo-entites. The hierarchical divisions (dashed lines) only partially correspond for the different data sources, cf. NUTS 1 regions and administrative divisions in GeoNames.

1. Initially we need an OSM data dump. We use Geofabrik,[16] a service to download extracts of OSM on a country level.
2. OSM provides different administrative levels for their relations, e.g., the relation which represents the states of a country, their subdivisions, and so on.[17] We use the alignment of these administrative levels with the previously introduced NUTS identifier to add the mappings to GeoNames: We perform lookups with the GeoNames labels of the NUTS 1, 2, and 3 regions at OSM's Nominatim service.[18] This service returns a set of potential candidate OSM relations for a given label. We disambiguate and select the correct relation (i.e. OSM region) by choosing the OSM relation at the same administrative/NUTS level as the corresponding GeoNames region.
3. Having the mapping for the countries' regions we again use OSM Nominatim to get the polygons for all regions. These polygons can be used to extract any street names, places, etc. from a OSM data extract.[19] For instance, using OSM Nominatim we get the polygon of the Viennese district "Leopoldstadt". We then use this polygon to extract all streets, POIs, and places in this district, and connect them to the corresponding GeoNames entity, i.e. add them to the hierarchy.

[16] http://download.geofabrik.de/, last accessed 2018-01-05
[17] http://wiki.openstreetmap.org/wiki/Tag:boundary%3Dadministrative
[18] http://nominatim.openstreetmap.org
[19] OSM provides a tool, Osmosis http://wiki.openstreetmap.org/wiki/Osmosis, to process polygons on OSM data dumps

Figure 1 displays the connections between the datasets; the bold arrows indicate the established links. As a basis serves the hierarchical structure of GeoNames. It already provides an administrative order of the top-level regions, however, this order is not necessary matching the European Union's NUTS classification. For instance, in Austria there are NUTS 2 regions which are not present in GeoNames (cf. AT1/Eastern Austria in Figure 1). We therefore use Wikidata to add the links between the NUTS classifications and the GeoNames regions.

The right column in Figure 1 displays the connections of GeoNames to national postal codes. Just like the NUTS identifier the links from GeoNames entries to postal codes are basically synonym labels, i.e. same-as relations (indicated by the two-way arrows). The OSM extracts, in contrast, are hierarchically connected to the GeoNames regions ("within"-relations), e.g., to the districts of the city of Vienna in the example in Figure 1.

3.2. Access Geo-Information in Open Data Portals

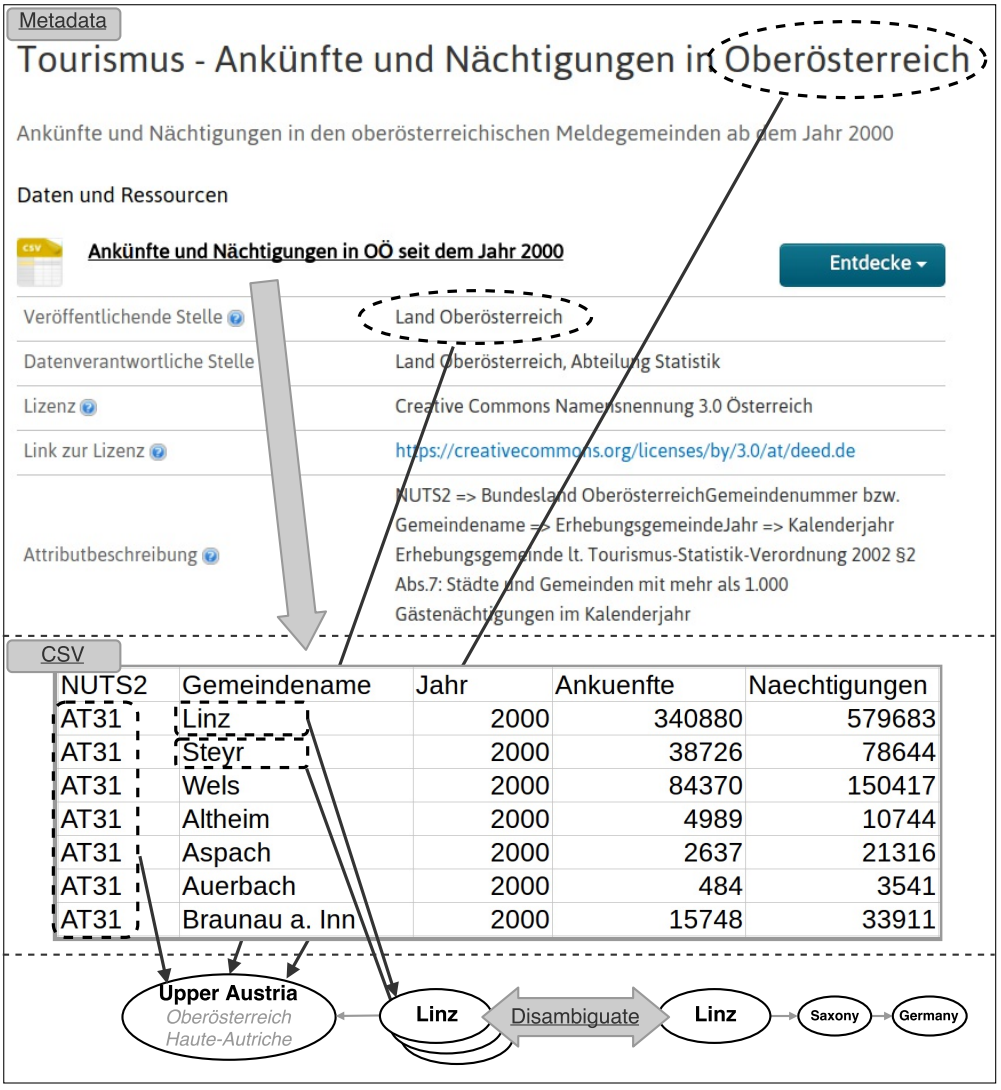

Fig. 2. Geo-information in metadata and CSVs. Example dataset from the Austrian data portal: https://www.data.gv.at/katalog/dataset/tourismus-ankunfte-und-nachtigungen-in-oberosterreich

In order to add geo-semantic labels to Open Data resources we use the metadata and CSV files from the data portals as signals. The metadata descriptions and datasets are provided by our Open Data Portal Watch framework

[13] which monitors and archives all the data from over 260 data portals, and provides APIs to retrieve their metadata descriptions in an homogenized way. Regarding the meta-information, we look into several available metadata-fields: we consider the title, description, the tags and keywords, and the publisher. For instance, the upper part of Figure 2 displays an example metadata description. It holds cues in the title and the publisher field (cf. "Veröffentlichende Stelle" - publishing agency) and holds a link to the actual dataset, a CSV file (cf. lower part in Figure 2), which we download and parse.

3.3. Labelling Algorithm

CSV columns. The CSVs are processed column by column. Currently, we do not consider any relations between two or multiple columns, i.e. the context of a CSV row, but of course there are many examples where this information could be further exploited, e.g., if addresses in CSVs are separated into dedicated columns for street, number, city, state, etc. In an updated version of our framework we want to detect these relations and use it as an additional cue for disambiguation of street/region values. The following algorithm is used to detect and label geo-entities in CSV columns:

1. Initially, the columns get classified based on simple regular expressions for NUTS identifier and postal codes. While the NUTS pattern is quite restrictive (NUTS start with two letters followed by zero to three numbers), the postal codes pattern has to be very general, potentially allowing many false positives. Basically, the pattern is designed to allow all codes in the integrated dataset (cf. Section 2) and to filter out other strings, words, and decimals.[20]

   (a) If we detect a potential NUTS column we try to map all values to existing NUTS identifier. If this is possible for a certain threshold (currently set to 90% of the values) we consider the values as NUTS identifier and add the respective semantic labels to them (using the GeoNames entities).
   (b) In case a column holds potential postal codes the algorithm again tries to map the values to existing postal codes. However, this time we restrict the set of codes to the originating country of the dataset. This again results in a set of semantic labels which we only accept with a threshold of 90%.

2. If the column consists of strings, i.e. some word or text, we first try to map the column values to GeoNames labels:

   (a) First, we collect all possible entity mappings for all column values, including any ambiguous labels. E.g., considering the CSV in Figure 2 we collect all GeoNames entries for "Linz" and likewise for the other values (Steyr, Wels, ...). For these we aggregate all their predecessors in the hierarchy, i.e., we recursively count all parent entities. E.g., we would count seven "Upper Austria" occurrences, assuming all the values have a mapping with this entity as their parent.
   (b) Then, to disambiguate a value which has multiple GeoNames candidates, we gather all predecessors of the candidates: we sum up the aggregated counts of these predecessors to resolve a mapping.
   For instance, in the example in Figure 2, on the one hand the German "Linz" candidate has a score of 2 because there was only one "Saxony" and one "Germany" in the aggregated predecessors counts (namely the predecessors of the German "Linz" candidate itself). On the other hand, the other candidate, the Austrian "Linz" has a much higher score (at least 14) because we counted "Upper Austria" and "Austria" as predecessors for all values in the column (Steyr, Wels, Altheim, ...). Therefore, by selecting the candidate with the highest score, we resolved the Austrian "Linz" as a mapping for the value.

3. If no GeoNames mapping was found (with a threshold of at least 50% mapped values) we try to instantiate the values with the OSM entries in our knowledge base. We apply the same algorithm as above to disambiguate any

[20] $\gamma(([A-Z\backslash d])\{2,4\}|([A-Z]\{1,2\}.)?\backslash d\{2,5\}(\backslash s[A-Z]\{2,5\})?(.[\backslash d]\{1,4\})?)\$$

multi-matches. As a preprocessing step, and in order to better parse addresses, we use the Libpostal library (cf. Section 2) to extract streets and place names from strings.
Additionally, to reduce the set of candidates, we use the entities found in the metadata (see paragraph below), in case there were any: We filter out those OSM candidate mappings which are within one of the regions detected in the metadata.

Metadata. The Metadata descriptions of the CSVs, which can be found on the Open Data portals, often give hints about the respective region covering the actual data. Therefore, we use this meta-information as another source and try to extract geo-entities from the titles, descriptions and publishers of the datasets:

1. As a first step, we tokenize the input fields, and remove any German and English stopwords.[21] Also, we split any words that are separated by dashes, underscores, semicolon, etc.
2. We group the input by word sequences of up to four words, i.e. all single words, groups of two words, ..., and run the previously introduced algorithm for mapping a set of values to the GeoNames labels (including the disambiguation step).

Figure 2 gives an example dataset description found on the Austrian data portal `data.gv.at`. The labelling algorithm extracts the geo-entity "Upper Austria" (an Austrian state) from the title and the publisher "Oberösterreich". The extracted geo-entities are added as additional semantic information to the indexed resource.

4. Indexed Datasets & Search Interface

Our showcase user interface currently contains CSV tables from four Open Data portals: We selected two Austrian and two German portals; respectively the governmental and non-governmental portals, cf. Section 4. Note, that the notion of *datasets* on Open Data portals (wrt. Section 4) usually groups a set of *resources*; for instance, typically a dataset groups resources which provide the same content in different file formats. A detailed description and analysis of Open Data portals' resources can be found in [13]. The statistics in Section 4 and the indexed CSVs are downloaded in the third week of December 2017. The differing numbers of *CSVs* and *indexed* documents in the table can be explained by offline resources, parsing errors, etc.

Table 1. Indexed data portals

portal	**datasets**	**resources**	*thereof* **CSVs**	**indexed**
data.gv.at	2399	9091	2794	2427
opendataportal.at	414	1061	473	442
govdata.de	19 711	56 584	14 542	5396
offenedaten.de	10 902	24 247	4408	3308

4.1. System Setup & Search Interface

The framework currently consists of three components: Firstly, the *geo-entities DB* where we store all labels from all the integrated datasets and their corresponding geo-entities. Since this DB is used as a look-up store, we use the NoSQL key-value database program MongoDB. It allows an easy integration of any data source and a very performant look-up of keys (e.g., labels, GeoNames IDs, postal codes, etc.).

Second, we use the *search engine* Elasticsearch to store and index the processed datasets. An Elasticsearch document corresponds to an indexed CSV in our framework and consists of all cell values of the table (arranged by columns), the potential geo-labels for a labelled column, metadata of the CSV (e.g., the data portal, title, publisher, etc.), and any additional geo-labels extracted from the metadata.

[21] Note, that currently the datasets are only from German-speaking countries and therefore no other languages are required.

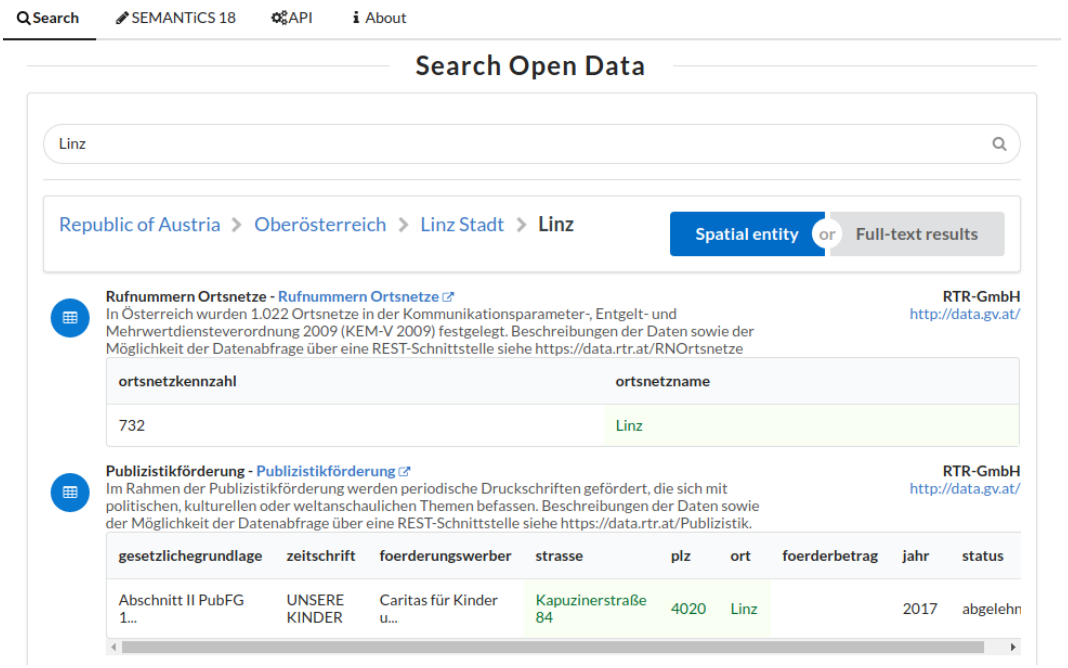

Fig. 3. Example query for the Austrian city "Linz" at the UI at http://data.wu.ac.at/odgraph/

The third component, displayed in Figure 3, is the showcase *web interface* which allows search queries for geo-entities but also full-text matches. Note, that in the current UI geo-entity search is implemented using auto-completion of the input (but only suggesting entries with existing datasets) and full-text querying is supported by using the "Enter"-key in the input form.

The screenshot in Figure 3 displays an example query for the Austrian city "Linz"; the green highlighted cells in the rows below show the annotated labels. Additionally, there is an API available to retrieve the search results, datasets, and RDF version of the annotations. All the source code for the labelling and user interface is on Github available: https://github.com/sebneu/geolabelling/

4.2. Manual Inspection and Evaluation of a Random Sample

In total, the Elasticsearch index currently consists of 11573 CSV documents and their metadata, harvested from the four Open Data portals. For 11028 documents we were able to add geo-labels based on their titles or publisher, which means that most of the indexed datasets (> 95%) got any geo-semantic information assigned. Regarding the CSVs' columns we were able to detect and link geo-entities in 3054 CSV documents (26%). For most of these columns we were able to add GeoNames identifiers (2850 documents), i.e. based on city/region names, postal codes, NUTS. Whereas, the number of documents where the algorithm labelled street names and places, i.e. OSM data, is rather small (306); the detailed numbers for each of the data portals are listed in Section 4.2.

Table 2. Number of CSVs with column and metadata labels.

	total	data.gv.at		opendataportal.at		govdata.de		offenedaten.de	
Columns	3054	717	*(30%)*	7	*(2%)*	1126	*(21%)*	1204	*(36%)*
GeoNames	2850	587		5		1105		1153	
OSM	306	185		3		75		43	
Metadata	11 028	2391	*(99%)*	441	*(99%)*	4895	*(91%)*	3301	*(99%)*

To get an idea about the performance and limitations of our geo-labelling approach we have randomly selected 100 datasets (using the Elasticsearch's built-in random function[22]) and performed two experimental evaluations: first selecting 40 random datasets to estimate the algorithms labelling performance, and second using a set of 60 datasets without labels to report potentially missing annotations. As for the main findings, in the following let us provide a short summary; all selected datasets and their assigned labels, along with a detailed description of the evaluation can be found at: http://data.wu.ac.at/odgraph/semantics

Eval. 1). In our first experimental evaluation a total of 40 datasets - ten datasets per indexed portal - were randomly selected. We manually categorize the datasets' labels by assessing the following dimensions: does a dataset contain any geo-spatial content, did the algorithm assign any correct or incorrect labels to columns and to the metadata descriptions, and is there some geo-information in the dataset that the algorithm did not detect.

Out of these 40 we identified 16 datasets that do not contain any geo-spatial data that could be mapped. All these 16 datasets are published on the two non-governmental portals opendataportal.at and offenedaten.de. From the remaining 24 datasets we identified in 17 datasets correct labels for cell values ($\sim$ 71%), while for 4 datasets we assigned incorrect OSM labels and also for 4 datasets incorrect GeoNames labels.

For 7 datasets we identified some content that would have been some additional geo-information but was not labelled by our approach. For instance, the dataset contains sub-district labels of a city which are not present in the knowledge graph, or city/region names that are embedded in text and/or use abbreviations (e.g. "Str" for "Straße"/street) and therefore were not correctly detected by our algorithm.

For 33 out of the 40 datasets we were able to derive a correct metadata label based on the title or publisher of the dataset ($\sim$ 83%). However, for 32 datasets we derived some (additional) incorrect metadata labels; note, that for now we allow multiple metadata labels. For instance, given the publisher "Stadt Wien" we link the two geo-entities "Vienna", the city of Vienna, Austria, and "Stadt", a German city in Saxony. An easy fix of this issue is to restrict the metadata labels to the origin country of the portal, however, we want our approach as general as possible so that we do not restrict the labels to national datasets only. Also, given that we use the labels in a search engine, these false positives can be considered as a minor issue, i.e. down-rated by an adequate ranking of the results, with the benefits of a more comprehensive result set.

Eval. 2). We inspect and report the potential false negative errors of our system, i.e. datasets where we did not assign any labels, by selecting another three random sets of tables: First, we select 20 random datasets where no column labels were assigned but a metadata label is available, second, 20 datasets where no metadata labels are available but column labels exist, and third, another 20 random datasets without assigned column or metadata labels.

None of the 60 sample datasets lack any geo-labelling based on the datasets' title and publisher. In particular, the 40 datasets without any assigned metadata labels do not provide any geo-information cues in their metadata descriptions. For 9 of the 60 datasets we identified columns with potential geo-data where the algorithm did not assign any labels. Particularly in the set of 20 datasets without any assigned metadata and column labels we found 7 candidates with missing labels.

These missed labels can be grouped into three basic error classes: (i) The corresponding entities are missing in the base knowledge graph so that our algorithm is not able to link the labels in the column context. To solve these errors we have to integrate more entities and/or find alternative names (e.g., by using the multi-lingual labels from Wikidata/DBpedia). (ii) The city/region names are embedded in text, or combined with other content in a single cell, e.g., the region type. These errors can be resolved by an improved pre-processing step for the cell values of the CSVs. (iii) The column contains very few labels, below the algorithm's threshold, or, similarly, the table consists of several sub-tables, where each sub-table has a geo-label as "title". To deal with this kind of errors we need an improved parsing algorithm which allows a better understanding of the table's structure, e.g., if the table is horizontally or vertically oriented.

[22] https://www.elastic.co/guide/en/elasticsearch/guide/current/random-scoring.html, last accessed 2018-01-08

5. Related Work

In the 2013 study [8] the authors give an overview of Semantic Web technologies in the geospatial domain and list Linked Data repositories and ontologies for geo-data. However, we were able to access only three of the listed repositories (including GeoNames). The 2012 project LinkedGeoData[16] resulted in a Linked Data resource, generated by converting OpenStreetMap to RDF and deriving a lightweight ontology from it. In [6] the authors describe their attempts to further connect GeoNames and LinkedGeoData, using string similarity measures and geometry matching. However, while LinkedGeoData is also listed in [8] as a geospatial Linked Data repository, it is currently not available online.

The GeoKnow project [11] is another attempt to provide and manage geospatial data as Linked Data. GeoKnow provides a toolset to process these datasets, including the storage, authoring, interlinking, and geospatially-enabled query optimization techniques. The project PlanetData (2010 to 2014) released an RDF mapping of the NUTS classifications[23] [7] using the GeoVocab vocabulary.[24] Unfortunately, the project does not include external links to GeoNames, or Wikidata.

GeoSPARQL [14] extends SPARQL to a geographic query language for RDF data. It defines a small ontology to represent geometries and connections between spatial regions (e.g., contains, part-of, intersects). While we do not provide this feature yet, we plan to make our base knowledge graph and RDFized linked data points from the CSVs available via a GeoSPARQL endpoint as part of future work.

Complementary to our approach, Open Addresses[25] is a global collection of address data sources. The manually collected and homogenized dataset consists of a total of 478M addresses; street names, house numbers, and post codes combined with geographic coordinates, harvested from governmental datasets of the respective countries. Currently, our system only uses street names from OSM, however, we plan to integrate Open Addresses in future work.

6. Conclusions & Future Work

Preceding to this work, we pointed out that Open Data tables – generally speaking – differ from Web tables in structure and content [12]: while HTML/Web tables typically have rich textual descriptions suitable for text-based entity linkage techniques, in Open Data we find a large portion of non-textual columns and missing or non human-readable headers. In earlier research [12] we proposed an approach to find and rank candidates of semantic labels for numerical columns to solve this issue. However, a main barrier there was the domain mismatch with the current LOD.

Inspired by, and complementary to these initial results, our research on how to better integrate Open Data tables into the LOD cloud, i.e., finding repositories with high domain overlap with the open datasets, lead to the popular geo-data knowledge bases GeoNames and OSM. In this work, we have introduced a framework to detect and establish links to geo-entities in datasets found on Open Data portals. To this end we have built up a base knowledge graph by integrating several geo-data repositories. We have used an heuristic labelling algorithm to link the information found in columns and metadata of the datasets to the knowledge graph. The labelled and indexed datasets harvested from four different data portals are accessible via an online search interface (`http://data.wu.ac.at/odgraph/`). Overall, our labelling approach was able to add geo-labels for most of the indexed datasets (>95%); for around 30% of the datasets we also assigned labels to columns in the tables. A manual inspection and evaluation of a random sample shows that the assigned geo-entites are sufficiently correct and complete, and the framework proves to be particularly useful in combination with an adequate search interface and ranking of the search results. More particularly, it can be considered a huge improvement of the current search functionalities of Open Data portals, which are mostly based on a full-text search over the metadata and do not support geo-based queries. In fact, Kacprzak et al. showed in [9], based on a query log analysis of open data portals, that particularly temporal and geospatial search queries require better support. To the best of our knowledge, this is the first work addressing a geo-semantic labelling and search of datasets based on a knowledge graph of geo-entities.

[23] http://nuts.geovocab.org/, last accessed 2018-01-05
[24] http://geovocab.org/, last accessed 2018-01-05
[25] https://openaddresses.io/, last accessed 2018-01-05

In future work, we plan to extend the number of covered portals and countries. Since our current approach is based on sufficiently rich and complete data sources (OSM, GeoNames) it can be easily extended to various other countries. To this end we use the Open Data Portal Watch framework [13] which monitors over 260 Open Data portals worldwide, including government portals of countries in Europe, South and North American and the Middle East, with a total of 1.1M datasets. Besides the geo-information of the datasets we identified a second, characterizing, dimension of datasets on Open Data portals: We also want to extract the datasets' temporal contexts. Similarly to the geospatial cues, temporal information in Open Data comes in various forms and granularity, e.g., as datetime/timespan information in the metadata indicating the validity of a dataset, or year/month/time information in CSV columns providing timestamps for data points or measurements. There are existing temporal taggers, e.g. the Heideltime framework [17], that provide datetime (and also time range) information from natural text.

References

[1] , 2013. ISO 3166-1, Codes for the representation of names of countries and their subdivisions. International Organization on Standardization. URL: https://www.iso.org/standard/63545.html.

[2] Attard, J., Orlandi, F., Scerri, S., Auer, S., 2015. A systematic review of open government data initiatives. Government Information Quarterly 32, 399 – 418. URL: http://www.sciencedirect.com/science/article/pii/S0740624X1500091X, doi:https://doi.org/10.1016/j.giq.2015.07.006.

[3] Butler, H., Daly, M., Doyle, A., Gillies, S., Schaub, T., Schaub, T., 2016. The GeoJSON Format. RFC 7946. URL: https://rfc-editor.org/rfc/rfc7946.txt, doi:10.17487/RFC7946.

[4] Esr, I., 1998. ESRI Shapefile Technical Description. Environmental Systems Research Institute, Inc. URL: http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf.

[5] Gurstein, M.B., 2011. Open data: Empowering the empowered or effective data use for everyone? First Monday 16.

[6] Hahmann, S., Burghardt, D., 2010. Connecting linkedgeodata and geonames in the spatial semantic web, in: 6th International GIScience Conference.

[7] Harth, A., Gil, Y., 2014. Geospatial data integration with linked data and provenance tracking, in: W3C/OGC Linking Geospatial Data Workshop, pp. 1–5.

[8] Janowicz, K., Scheider, S., Adams, B., 2013. A geo-semantics flyby, in: Reasoning web. Semantic technologies for intelligent data access. Springer, pp. 230–250.

[9] Kacprzak, E., Koesten, L.M., Ibáñez, L.D., Simperl, E., Tennison, J., 2017. A query log analysis of dataset search, in: Cabot, J., De Virgilio, R., Torlone, R. (Eds.), Web Engineering, Springer International Publishing, Cham. pp. 429–436.

[10] Kubler, S., Robert, J., Neumaier, S., Umbrich, J., Traon, Y.L., 2017. Comparison of metadata quality in open data portals using the analytic hierarchy process. Government Information Quarterly URL: http://www.sciencedirect.com/science/article/pii/S0740624X16301319, doi:https://doi.org/10.1016/j.giq.2017.11.003.

[11] Lehmann, J., Athanasiou, S., Both, A., García-Rojas, A., Giannopoulos, G., Hladky, D., Le Grange, J.J., Ngomo, A.C.N., Sherif, M.A., Stadler, C., et al., 2015. Managing geospatial linked data in the geoknow project.

[12] Neumaier, S., Umbrich, J., Parreira, J.X., Polleres, A., 2016a. Multi-level semantic labelling of numerical values, in: International Semantic Web Conference, Springer. pp. 428–445. URL: https://doi.org/10.1007/978-3-319-46523-4_26.

[13] Neumaier, S., Umbrich, J., Polleres, A., 2016b. Automated quality assessment of metadata across open data portals. J. Data and Information Quality 8, 2:1–2:29. URL: http://doi.acm.org/10.1145/2964909, doi:10.1145/2964909.

[14] Perry, M., Herring, J., 2012. OGC GeoSPARQL - A geographic query language for RDF data. OGC Implementation Standard. Sept .

[15] Posada-Sánchez, M., Bischof, S., Polleres, A., 2016. Extracting geo-semantics about cities from openstreetmap., in: SEMANTiCS (Posters, Demos, SuCCESS).

[16] Stadler, C., Lehmann, J., Höffner, K., Auer, S., 2012. Linkedgeodata: A core for a web of spatial open data. Semantic Web 3, 333–354.

[17] Strötgen, J., Gertz, M., 2013. Multilingual and cross-domain temporal tagging. Language Resources and Evaluation 47, 269–298. doi:10.1007/s10579-012-9179-y.